801681_1.DOC
DJT/BMN/lb
May 13, 2008

OIPE
MAY 15 2008
PATENT & TRADEMARK OFFICE

PATENT APPLICATION
Docket No.: 3721.1004-005 (EQLC-P01-006)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	G. Paul Koning, Peter C. Hayden, Paula Long, and Daniel E. Suman		
Application No.:	10/762,985	Group:	2157
Filed:	January 21, 2004	Examiner:	Nano, Sargon N.
Confirmation No.:	5999		
For:	Client Load Distribution		

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450 Alexandria, VA 22313-1450, or is being facsimile transmitted to the United States Patent and Trademark Office on:

5/13/2008 Loretta A. Boudreau

Date Signature

LORETTA A. BOUDREAU

Typed or printed name of person signing certificate

NOTIFICATION OF REQUEST FOR REMOVAL OF SMALL ENTITY STATUS

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicant requests removal of small entity status for this application pursuant to 37 CFR § 1.27(g)(2).

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By

David J. Thibodeau, Jr.
Registration No.: 31,671
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133
Date: 5/13/08

807188_1.DOC
DJT/BMN/lb/(aeg)
May 13, 2008

OIPE
MAY 15 2008
PATENT & TRADEMARK OFFICE

PATENT APPLICATION
Docket No.: 3721.1004-005 (EQLC-P01-006)

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	G. Paul Koning, Peter C. Hayden, Paula Long, and Daniel E. Suman		
Application No.:	10/762,985	Group:	2157
Filed:	January 21, 2004	Examiner:	Nano, Sargon N.
Confirmation No.:	5999		
For:	CLIENT LOAD DISTRIBUTION		

CERTIFICATE OF MAILING OR TRANSMISSION

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, or is being facsimile transmitted to the United States Patent and Trademark Office on:

5/13/2008 Loretta A. Boudreau
Date Signature

LORETTA A. BOUDREAU
Typed or printed name of person signing certificate

APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is submitted pursuant to the Notice of Appeal received in the U.S. Patent and Trademark Office on November 27, 2007, and in support of the appeal from the final rejection set forth in the Office Action mailed on August 9, 2007. The fee for filing a brief in support of an appeal is enclosed. A Petition for Extension of Time and the appropriate fee are being filed concurrently.

05/15/2008 NNGUYEN1 00000025 10762985
01 FC:1402 510.00 OP

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST

The real party in interest is Equallogic Inc., now located at 110 Spit Brook Road, Building ZKO2, Nashua, New Hampshire 03062. Equallogic Inc. is the Assignee of the entire right, title and interest in the subject application, by virtue of an Assignment recorded on January 21, 2004 at Reel 014930, Frames 0925-0928.

II. RELATED APPEALS AND INTERFERENCES

Appellants, the undersigned Attorney and Assignee are not aware of any related appeals or interferences which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1, 3-8, and 10-13 have been finally rejected, and a copy appears in the Appendix of this Brief. Claims 1, 3, 6-8, and 10 were amended and claims 2 and 9 were canceled in the Amendment filed on May 18, 2007. Claims 4-5, 11-12 and 13 therefore appear as originally filed. Claims 1, 3-8, and 10-13 are on appeal. A copy of the appealed claims appears in the Claims Appendix of this Brief.

IV. STATUS OF AMENDMENTS

No Amendments have been filed subsequent to the Final Rejection.

V. SUMMARY OF CLAIMED SUBJECT MATTER

A. Claim 1

Claim 1 is generally directed to a system for managing a set of connections between a plurality of clients and a plurality of servers based on system load.

In that system, a plurality of storage servers have a set of resources partitioned thereon. That is, portions of a given resource are stored across various storage servers in the system. (*See,* e.g., Summary of the Invention; Fig. 3; page 12, lines 1-16.)

Each server has a respective load monitor process capable of communicating with load monitor processes in other servers for generating a measure of system load, and a client load on each of the plurality of servers. (*See,* e.g., Summary of the Invention; Figs. 3 and 5; and page 16, lines 4-19.)

A client distribution process, responsive to the system load, is capable of repartitioning the set of client connections for distributing client load by moving at least one client connection from a first server to a second server. (*See,* e.g., Summary of the Invention; Figs. 3 and 5; and page 13, lines 1-4.)

More particularly, the claimed "plurality of storage servers having a set of partitioned resources thereon can be found in the description of an example embodiment in the specification. Fig. 1 depicts a client 12 communicating with a plurality of servers, 161, 162 and 163 in a server group 16. Clients 12 may access resources partitioned across the server group 16. Accordingly, each of the clients 12 send access requests to the server group 16. The clients typically act independently, and as such, the client load placed on the server group 16 will vary over time. In a typical operation, a client 12 will contact one of the servers, for example, server 161 in the group 16, to access a resource, such as a data block, page, file, database, application, or other resource. The contacted server 161 itself may not hold or have control over the requested resource. In a preferred embodiment, the server group 16 is configured to make all the partitioned resources available to the client 12 regardless of the server that initially receives the request. For illustration, the diagram shows two resources, one resource 18 that is partitioned over all three servers, servers 161, 162, 163, and another resource 17 that is partitioned over two of the three servers. In an example application where the system 10 is a block data storage system, each resource 18 and 17 may represent a partitioned block data volume.

More particularly, the claimed "load monitor process capable of communicating with load monitor processes in other servers for generating a measure of system load, and a client load on each of the plurality of servers" is also explained with reference to the example embodiment. In the Specification at Fig. 5 and page 14, line 13 to page 15, line 14, each server is said to include a routing table (depicted as routing tables 20A, 20B and 20C) and a load monitor process, 22A, 22B and 22C respectively. As further described at page 16, lines 5-16, the load monitor processes 22A, 22B and 22C each observe the request patterns arriving at their respective equivalent servers to determine whether patterns or requests from clients 12 are being forwarded to the server group 16 and whether these patterns can be served more efficiently or reliably by a different arrangement of client connections to the several servers. In one embodiment, the load monitor processes 22A, 22B and 22C monitor client requests coming to their respective equivalent servers. In one embodiment, the load monitor processes each build a table representative of the different requests that have been seen by the individual request monitor processes. Each of the load monitor processes 22A, 22B and 22C are also capable of communicating between themselves for the purpose of building a global database of requests seen by each of the equivalent servers.

More particularly, the claimed "client distribution process, responsive to the system load", is also capable of "repartitioning the set of client connections for distributing client load by moving at least one client connection". That is, any initial client-server assignment for a new connection based on load (See, e.g., Specification at page 18, line 18 to page 19, line 8) may subsequently not yield the best overall or server-specific performance. For example, in Fig. 5, a client distribution process 30 moves the connection for client 3 (12C) connection from server 161 to server 162 based on load measured by a distributed load monitoring process. (See, e.g., Specification at page 20, lines 14-17.) Once the connection is moved, server 162 services requests for client 3 (12C) requests (which may involve accessing other servers for portions of the resource that are not stored on server 162).

B. Claim 10

Claim 10 depends from claim 1 and adds a limitation that the client distribution process adaptively distributes client connections across the plurality of servers as a function of dynamic variations in measured system load.

Support for this claim can be found, with reference to the description of an example embodiment, which is described as a system that periodically measures and then "redistributes" existing client connections based on the observed load. As explained in the Specification at page 17, line 25- page 18, line 15). the client distribution process 30 for server group 16 can make a determination that overall efficiency may be improved by redistributing client load from its initial condition. Considerations that drive this load balancing decision may vary and some examples are the desire to reduce routing: for example if one server is the destination of a significantly larger fraction of requests than the others on which portions of the resource (e.g., volume) resides, it may be advantageous to move the connection to that server. Or to further have balancing of server communications load: if the total communications load on a server is substantially greater than that on some other, it may be useful to move some of the connections from the highly loaded server to the lightly loaded one, and balancing of resource access load (e.g., disk I/O load)--as preceding but for disk I/O load rather than comm load. This optimization process that involves multiple dimensions, and the specific decisions made for a given set of measurements may depend on administrative policies, historical data about client activity, the capabilities of the various servers and network components, etc. To this end, FIG. 5 depicts this redistribution of client load by illustrating a connection 325 (depicted by a dotted bi-directional arrow) between client 12C and server 162. It will be understood that after redistribution of the client load, the communication path between the client 12C and server 161 may terminate.

C. Claim 12

Claim 12 depends from claim 1 and adds that a storage device provides at least one volume of storage partitioned across the plurality of servers.

At page 7, lines 7-9 it is explained that for the purpose of illustration, the claimed invention is described with reference to systems and methods for managing the allocation of data blocks across a partitioned volume of storage. At page 10, lines 2-4, it is also stated that in the

example of system 10 being a block data storage system, each resource 18 and 17 may represent a partitioned block data volume.

D. Claim 13

Claim 13 is directed to a storage area network ("SAN") that comprises a plurality of servers, wherein each server is configured as the server of claim 1.

Support for this claim can thus be found in the sections of the specification referenced above for claim 1. In addition, at page 10, lines 5-7, it is stated that the embodiment of FIG. 1, the server group 16 provides a block data storage service that may operate as a storage area network (SAN) comprised of a plurality of equivalent servers, servers 161, 162 and 163. There are further references to an SAN embodiment, such as at page 15, lines 5-8 where it is stated that "..the server group may be a SAN, or part of a SAN, wherein each of the equivalent servers 161, 162, 163 has an individual IP address that may be employed by a client 12 for accessing that particular equivalent server on the SAN".

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1, 3-8, and 10-13 were rejected under 35 U.S.C. § 102(e) by the Examiner as supposedly being anticipated by O'Neil et al. (U.S. Patent Number 6,128,279) (hereinafter "O'Neil").

VII. ARGUMENT

A. The Examiner Has Failed to State a *Prima Facie* Case of Anticipation Because the Cited Art Reference Fails to Teach Each and Every Element of the Claimed Invention

Claims 1, 3-8, and 10-13 were rejected under 35 U.S.C. § 102(e) by the Examiner as supposedly being taught by O'Neil et al. (U.S. Patent 6,128,279).

1. The O'Neil Patent

In brief, O'Neil describes a plurality of network servers which provide load balancing on a peer-to-peer basis. When any of the servers receives a request, the server either processes the request or routes the request to one of its peers depending on their respective loads and/or on the contents of the request. By implementing load balancing directly on the servers, the need for dedicated load balancing hardware is reduced, as are the disadvantages resulting from such hardware. Thus, for example, because each server has the capability to perform load balancing, access to a Web site managed by the server is not subject to a single point of failure. (*See,* O'Neil, column 3, lines 18-30.)

2. Claims 1, 3-8, and 10-13 are Novel because O'Neil Does Not Have the Set of Resources Partitioned Thereon

Title 35 U.S.C. §102(e) provides:

> A person shall be entitled to a patent unless - the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Anticipation under 35 U.S.C. §102 requires identical disclosure of the claimed invention in the prior art. (*See* Gechter v. Davidson, 116 F.3d 1454, 1457, 43 USPQ2d 1030, 1032 (Fed. Cir. 1997)) ("Under 35 U.S.C. §102, every limitation of a claim must identically appear in a single prior art reference for it to anticipate the claim.") "Every element of the claimed invention must be literally present, arranged as in the claim." (*See* Merck & Co. v. Teva Pharms. USA, Inc., 347 F.3d 1367, 1372 (Fed. Cir. 2003); and Richardson v. Suzuki Motor Co., Ltd., 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989)). As also provided in MPEP 2131, "[a] claim is anticipated only if each and every element as set forth in the claim is found."

The Examiner appears to have overlooked Appellants' claimed limitation of "a plurality of storage servers having the set of resources partitioned thereon," as the Examiner provides no

support in the rejection for it. Nowhere does O'Neil mention or suggest a plurality of storage servers having the set of resources partitioned thereon. For this reason alone the anticipation rejection should be withdrawn.

Referring to the industry standard definition for "partition," the term means "A logically distinct portion of memory or a storage device that functions as though it were a physically separate unit." (*See* Exhibit A, "Computer Dictionary," Microsoft Press, Fifth Edition, 2002.) Keeping this definition of "partition" in mind, the operation of Appellants' partitioned-resource storage system can be understood from the illustrative embodiment of Figs. 1 and 3. Fig. 1 depicts client 12 communicating with a plurality of servers, 161, 162 and 163. In operation, clients 12 may need resources partitioned across the server group 16. Accordingly, each of the clients 12 will send requests to the server group 16. The clients typically act independently, and as such, the client load placed on the server group 16 will vary over time. In a typical operation, a client 12 will contact one of the servers, for example, server 161 in the group 16, to access a resource, such as a data block, page, file, database, application, or other resource. The contacted server 161 itself may not hold or have control over the requested resource. However, in a preferred embodiment, the server group 16 is configured to make all the partitioned resources available to the client 12 regardless of the server that initially receives the request. For illustration, the diagram shows two resources, one resource 18 that is partitioned over all three servers, servers 161, 162, 163, and another resource 17 that is partitioned over two of the three servers. In an example application where the system 10 is a block data storage system, each resource 18 and 17 may represent a partitioned block data volume.

Referring to Fig. 3, each server in the group 16 contains a routing table 165 for each volume, with the routing table 165 identifying the server on which a specific page of a specific volume can be found. For example, when the server 161 receives a request from a client 12 for volume 3, block 93847, the server 161 calculates the page number (page 11 in this example for the page size of 8192) and looks up in the routing table 165 the location or number of the server that contains page 11. If server 163 contains page 11, the request is forwarded to server 163, which reads the data and returns the data to the server 161. Server 161 then sends the requested data to the client 12. In other words, the response is always returned to the client 12 via the same server 161 that received the request from the client 12. It is transparent to the client 12 to which

servers 161, 162, 163 are connected. Instead, the client only sees the servers in the server group 16 and requests the resources of the server group 16.

O'Neil does not disclose anything like Appellants' claimed system. O'Neil is not partitioning resources across storage servers and none of O'Neil's servers provide access to a resource having a portion partitioned thereon (and the remainder on other servers). Thus, O'Neil is wholly inapposite to Appellants' claimed system.

3. Claims 1, 3-8, and 10-13 are Novel because O'Neil Does Not Disclose a Process in which Client Connections are Mov[ed] from a First Server ... to a Second Server

Appellants' claimed invention further includes elements for for redistributing a set of connections between a plurality of clients and a plurality of servers that managed the partitioned resource, based on observed system load. The claimed system thus includes a client distribution process, responsive to the system load, and capable of repartitioning the set of client connections for distributing client load by moving at least one client connection from a first server of the plurality of servers to a second server of the plurality of servers. The operation of Appellants' connection management in a partitioned-resource storage system can be understood from the illustrative embodiment of Fig. 5. As an initial condition for this example, three clients have a connection established with server 161, and two clients have a connection established with server 163. No clients have a connection established with server 162. Each server 161, 162, and 163 is responsible for different data blocks, pages, etc. of a resource partitioned across the group, and is capable of servicing client requests. In such a system, a client's requests are serviced by the server with which the client has an established connection. If a server does not have the portion of a resource relevant to a client's request (e.g., the relevant block, page, etc.), the server will communicate with another server within the group to obtain the necessary portion and return a reply to the client.

The client-server connections in such a system are long-lived and typically last across periods of time during which different system and client loads are experienced by the group and respective servers. As a result, any initial client-server assignment for a new connection based on load (*See,* e.g., Specification at page 18, line 18 to page 19, line 8) may subsequently not yield the best overall or server-specific performance. To address this issue, the system periodically "redistributes" existing client connections based on load. For example, in Fig. 5, a client

distribution process 30 moves the connection for client 3 (12C) connection from server 161 to server 162 based on load measured by a distributed load monitoring process. (*See,* e.g., Specification at page 20, lines 14-17.) Once the connection is moved, server 162 services requests for client 3 (12C) requests (which may involve accessing other servers for portions of the resource that are not stored on server 162).

The Examiner points to O'Neil at column 4, lines 1-9, as supposedly teaching "a client distribution process, responsive to the system load, and capable of repartitioning the set of client connections for distributing client load by moving at least one client connection from a first server of the plurality of servers to a second server of the plurality of servers." The Appellants' respectfully disagree. *O'Neil does not actually disclose moving any established client connections between servers*. Rather, O'Neil at column 4, lines 1-9, merely teaches that the client requests are rerouted to particular servers. Elsewhere O'Neil actually makes it clear that this re-routing of requests does not involve moving a connection from one server to another, but rather involves redirecting the client to the new server. For example, "In the invention, routing is performed by sending a command from load balancing module 17 to a requestor instructing the requestor" (e.g, by sending an instruction to the the client) "to send the request to a designated server." (*See* O'Neil column 7, lines 26-30; column 8, lines 32-33; column 8, lines 51-53; and column 9, lines 2-4.)

Not surprisingly, O'Neil does not move established connections, since O'Neil discloses an embodiment that is particularly concerned with World Wide Web ("WWW") servers. (*See,* e.g., column 4, line 62 to column 5, line 6.) To this end, the WWW is based on the Hypertext Transfer Protocol ("HTTP") to carry requests from a browser to a Web server and to transport pages from the Web servers back to the requesting browser. The connections in the HTTP request/response protocol are short lived. Thus, the connections in O'Neil are closed after a single request-response pair. There is no moving of connections in O'Neil as required in Appellants' claimed invention. In a partitioned storage system, connections may last much longer and across disparate loading conditions. Therefore, the capability of moving connections from one server to another is desirable.

Accordingly, claim 1 calls for a client distribution process, responsive to the system load, and capable of repartitioning the set of client connections for distributing client load by moving

at least one client connection from a first server of the plurality of servers to a second server of the plurality of servers.

O'Neil fails to teach, suggest, or otherwise make obvious "a plurality of storage servers having the set of resources partitioned thereon" and "a client distribution process, responsive to the system load, and capable of repartitioning the set of client connections for distributing client load by moving at least one client connection from a first server of the plurality of servers to a second server of the plurality of servers." Thus, O'Neil does not provide an identical disclosure of at least two features of the claimed invention. Every claim either recites these limitations, or contains these limitations through dependency. Therefore, Appellants' respectfully submit that the Examiner has failed to make out a prima facie case under 35 U.S.C. §102.

Claim 10 is certainly patentable for additional reasons. Specifically, claim 10 requires that the client distribution process adaptively distributes client connections across the plurality of servers as a function of dynamic variations in measured system load. O'Neil does not adaptively distribute client connections as a function of dynamic variations. Rather, O'Neil load balancing technique is based on a predetermined level. For example, a load balancing module determines the load of server 7 that is currently processing, and the capacity remaining therein. In a preferred embodiment, the predetermined level is 50%, meaning that server 7 is operating at 50% capacity. The value for the first predetermined level may be reprogrammed periodically. If server 7 is not processing a load that exceeds the first predetermined level, the network request is processed and a response thereto is output via the appropriate channel. However, if server 7 is processing a load that exceeds the first predetermined level, a load module determines whether the load being processed by server 7's peer are less than the load on server 7 by a differential exceeding a second predetermined level. This second predetermined level is set at 20%, but can also be reprogrammed. (*See,* e.g., O'Neil column 6, lines 11-61.) In contrast, Appellants' distribution process adaptively distributed connections as a function of dynamic variations in measured system load. There is no set predetermined level in Appellants' invention, but rather the process is adaptively distributed based on dynamic variations. The dynamic variations involves multiple dimensions, and the specific decisions made for a given set of measurements may depend on administrative policies, historical data about client activity, the capabilities of the various servers and network components, etc.

Claim 12 is also certainly patentable for additional reasons. In particular, a storage device providing at least one volume of storage partitioned across the plurality of servers. O'Neil has no discussion of a storage device providing at least one volume of storage partitioned across the plurality of servers. Although O'Neil has several back-end servers, these back-end severs do not have at least one volume of storage that is partitioned across the plurality of servers.

Claim 13 is directed to a storage area network ("SAN") that comprises a plurality of servers, wherein each server is configured as the server of claim 1. Claim 13 is thus patentable for the same reasons stated above for claim 1 and additional reasons. More specifically, nowhere does O'Neil teach, mention, or even suggest the use of his servers to provide a Storage Area Network (SAN) – again, not surprising given his stated purpose of providing a web server function.

Appellants respectively submit that all the remaining claims in the application are in condition for allowance based on their dependency on independent claim 1. The cited references failed to disclose all claim limitations as required for a rejection under 35 U.S.C. §102(e).

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By____________________________
David J. Thibodeau, Jr.
Registration No.: 31,671
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133
Dated: 5/12/08

CLAIMS APPENDIX

1. A system for managing a set of connections between a plurality of clients and a plurality of servers based on system load, comprising:

   a plurality of storage servers having the set of resources partitioned thereon, each server having

   a load monitor process capable of communicating with other load monitor processes for generating a measure of system load, and a client load on each of the plurality of servers; and

   a client distribution process, responsive to the system load, and capable of repartitioning the set of client connections for distributing client load by moving at least one client connection from a first server of the plurality of servers to a second server of the plurality of servers.

2. (Cancelled).

3. A system according to claim 1, further comprising:

   a load distribution process for determining resource loads when moving client connections among servers.

4. A system according to claim 1, further comprising:

   a client allocation process for causing a client to communicate with a selected one of said plurality of servers.

5. A system according to claim 1, further comprising:

   a client allocation process for distributing incoming client requests across said plurality of servers.

6. A system according to claim 1, wherein the client distribution process includes a round robin distribution process.

7. A system according to claim 1, wherein the client distribution process includes a client redirection process.

8. A system according to claim 1, wherein the client distribution process includes a disconnect process for dynamically disconnecting a client from a first server and reconnecting to a second server.

9. (Cancelled).

10. A system according to claim 1, wherein the client distribution process adaptively distributes client connections across the plurality of servers as a function of dynamic variations in measured system load.

11. A system according to claim 1, further comprising:

    a storage device for providing storage resources to the plurality of clients.

12. A system according to claim 1, further comprising:

    a storage device for providing at least one volume of storage partitioned across the plurality of servers.

13. A storage area network, comprising:

    a plurality of servers each configured as a server of claim 1.

EVIDENCE APPENDIX

A. Definition of "partition" in "Computer Dictionary," Microsoft Press, Fifth Edition, 2002.

RELATED PROCEEDINGS APPENDIX

NONE

parrallaxing *n.* A 3-D animation technique, often used by computer game developers, where backgrounds are displayed using different levels of speed to achieve realism. For example, distant levels move at a slower speed than closer levels, thereby giving the illusion of depth. *See also* animation.

parse *vb.* To break input into smaller chunks so that a program can act upon the information.

parser *n.* An application or device that breaks data into smaller chunks so that an application can act on the information. *See also* parse.

partition *n.* **1.** A logically distinct portion of memory or a storage device that functions as though it were a physically separate unit. **2.** In database programming, a subset of a database table or file.

Partition Boot Sector *n.* The first sector in the system (startup) partition of a computer's bootable hard disk, or the first sector of a bootable floppy disk. On an x86-based computer, the Partition Boot Sector is read into memory at startup by the Master Boot Record. It is the Partition Boot Sector that contains the instructions required to begin the process of loading and starting the computer's operating system. *See also* Master Boot Record, partition table.

partition table *n.* A table of information in the first sector of a computer's hard disk that tells where each partition (discrete portion of storage) on the disk begins and ends. The physical locations are given as the beginning and ending head, sector, and cylinder numbers. In addition to these "addresses," the partition table identifies the type of file system used for each partition and identifies whether the partition is bootable—whether it can be used to start the computer. Although it is a small data structure, the partition table is a critical element on the hard disk.

partnership *n.* The settings on a desktop computer and Windows CE device that allow information to be synchronized, as well as copied or moved between the computer and device. The mobile device can have partnerships with up to two desktop computers. *See also* synchronization (definition 6).

Pascal *n.* A concise procedural language designed between 1967 and 1971 by Niklaus Wirth. Pascal, a compiled, structured language built upon ALGOL, simplifies syntax while adding data types and structures such as subranges, enumerated data types, files, records, and sets. *See also* ALGOL, compiled language. *Compare* C.

pASP *n. See* pocket Active Server Pages.

pass[1] *n.* In programming, the carrying out of one complete sequence of events.

pass[2] *vb.* To forward a piece of data from one part of a program to another. *See also* pass by address, pass by value.

pass by address *n.* A means of passing an argument or parameter to a subroutine. The calling routine passes the address (memory location) of the parameter to the called routine, which can then use the address to retrieve or modify the value of the parameter. *Also called:* pass by reference. *See also* argument, call[1]. *Compare* pass by value.

pass by reference *n. See* pass by address.

pass by value *n.* A means of passing an argument or a parameter to a subroutine. A copy of the value of the argument is created and passed to the called routine. When this method is used, the called routine can modify the copy of the argument, but it cannot modify the original argument. *See also* argument, call[1]. *Compare* pass by address.

passivation *n.* In Sun Microsystems's J2EE network platform, the process of "turning off" an enterprise java bean (EJB) by caching it from memory to secondary storage. *See also* Enterprise JavaBeans, J2EE. *Compare* activation.

passive hub *n.* A type of hub used on ARCnet networks that passes signals along but has no additional capability. *See also* ARCnet. *Compare* active hub, Intelligent hub.

passive-matrix display *n.* An inexpensive, low-resolution liquid crystal display (LCD) made from a large array of liquid crystal cells that are controlled by transistors outside of the display screen. One transistor controls an entire row or column of pixels. Passive-matrix displays are commonly used in portable computers, such as laptops and notebooks, because of their thin width. While these displays have good contrast for monochrome screens, the resolution is weaker for color screens. These displays are also difficult to view from any angle other than straight on, unlike active-matrix displays. However, computers with passive-matrix displays are considerably cheaper than those with active-matrix screens. See the illustration. *Also called:* dual-scan display. *See also* liquid crystal display, supertwist display, transistor, twisted nematic display. *Compare* active-matrix display.

P

EXHIBIT

A